UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/601,015	06/19/2003	Eric Teh Gim Aik	ALTRP085/ A880	7118

51501 7590 08/08/2005

BEYER WEAVER & THOMAS, LLP
ATTN: ALTERA
P.O. BOX 70250
OAKLAND, CA 94612-0250

EXAMINER	
ROSSOSHEK, YELENA	
ART UNIT	**PAPER NUMBER**
2825	

DATE MAILED: 08/08/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/601,015	AIK, ERIC TEH GIM	
	Examiner	Art Unit	
	Helen Rossoshek	2825	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *19 June 2003*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-49* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☐ Claim(s) *1-3,7-14,16,17,21-24,26-41 and 43-49* is/are rejected.

7)☒ Claim(s) *4-6,15,18-20,25 and 42* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *19 June 2003* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *06/19/2003*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. This office action is in response to the Application 10/601,015 filed 06/19/2003.

2. Claims 1-49 are pending in the Application.

Claim Objections

3. Claim 29 is objected to because of the following informalities:

claim 29 line 3 after "rules" delete "; and" insert --.--

Appropriate correction is required.

Claim Rejections - 35 USC § 102

4. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

5. Claims 1-3, 7-14, 16, 17, 21-24, 26-41 and 43-49 are rejected under 35 U.S.C. 102(b) as being anticipated by Fairbanks (US Patent 6,182,020).

With respect to claim 1, 16, 31 Fairbanks teaches a method (col. 1, ll.65-67), apparatus and program for performing design rule checking on an electronic design within software base system such as MAX-PLUS available from Altera including computer design tools as software-implemented tool for assisting in the design of logic circuits providing technique for checking a user's design against a set of design rules (col. 1, ll.17-20, ll.21-26); the method comprising: receiving a netlist of the electronic design, the netlist having a plurality of nodes as shown on the Fig. 1, wherein the netlist

is stored in the file 10 and received by converting into a simulator logic netlist 14 (col. 1, ll.30-33; col. 3, ll.46-49); extracting information from each of the plurality of nodes in the netlist, whereby the extracted information is available to a set of predefined rules within simulator netlist extraction (SNF) wherein the subset of design rules is available for each device (node) in the extracted nodes of the netlist and executed to check a violation of the rules (col. 4, ll.7-11) also as shown in a portion of the program code (col.7, ll.42-45) wherein the extracted information for each device D (node) in netlist N is demonstrated; applying the set of predefined rules to the previously extracted information of the plurality of nodes in the list as shown on the Fig. 6 (as a flow diagram of a system for designing and testing logic circuitry) within a step of applying a design rule to the logic design, wherein defining logic design is done by extracting the information from netlist as shown on the Fig. 1, also within an ability of the method to create a structure to define a node in the logic tree captured in the netlist (col. 19, ll.49-55); and determining whether any of the predefined rules have violated by any of the plurality of nodes, wherein the applying operation is performed by a rule checking routine, which checks the electronic design as shown on the Fig. 6 within a step "does the logic design violate the rule?" (col. 1, ll.65-67).

With respect to claims 35 and 46 Fairbanks teaches: a method (col. 1, ll.65-67) and computer program product of performing design rule checking on a netlist representation of an electronic design within software base system such as MAX-PLUS available from Altera including computer design tools as software-implemented tool for assisting in the design of logic circuits providing technique for checking a user's design

against a set of design rules (col. 1, ll.17-20, ll.21-26); comprising (a) receiving a plurality of design rules (col. 3, ll.46-49) specifying constraints on the properties of particular nodes in a netlist, wherein each rule specifies a logical combination of constraints on properties of particular nodes in the netlist as shown on the example of the programming code on the column 1, lines 35-49 wherein variety of specific logical combination of constraints, which applied to the extracted information of the nodes from the netlist as shown in a portion of the program code (col.7, ll.42-45) wherein the extracted information for each device D (node) in netlist N is demonstrated, without including functionality for extracting the properties from the netlist when aggressive MPLD conversion is chosen, every rule is applied (col. 4, ll.38-40); (b) at a first node of the netlist, employing a generic routine to execute a first design rule from the plurality of design rules as shown on the example of the programming code shown on the column 17, lines 1-29 wherein design rule is applied to each nodes in the netlist; (c) at the first node of the netlist, employing the generic routine to execute a second design rule from the plurality of design rules and determine whether properties of the first node violate the second design rule (col. 24, ll.6-12).

With respect to claims 2, 3, 7-14, 17, 21-24, 26-30, 32-34, 36-41, 43-45, 47-49 Fairbanks teaches:

claims 2, 17, 32: the applying operation comprises: recognizing whether any of the predefined rules require extracted information from a neighboring node, the neighboring node being different from a node currently under consideration as shown in the program code on the column 17:

```
drc_check_macrofunctions( )
{
read in the device family for this compilation
read in the database of macrofunctions
for all nodes in the hierarchy tree
{
look up the record for this macrofunction
see if there are more efficient macrofunctions listed for
this entry
if so, inform the user of the better possible choice
}
```

within a hierarchical structure of the node tree and checking this nodes in the hierarchical order by DRC; and identifying the neighboring node for comparing under the rule checking routine if any of the predefined rules require extracted information from the neighboring node, wherein the recognizing and identifying operations are collectively performed by a traverser routine as shown above in the program code for design rule checking and looking for neighboring nodes in the hierarchical order checking all nodes going from one node to another (col. 17, ll.-28);

claim 3: the recited operations are each done automatically, without user intervention within software base system such as MAX-PLUS available from Altera including computer design tools as software-implemented tool for assisting in the design of logic circuits providing technique for checking a user's design against a set of design rules (col. 1, ll.17-20, ll.21-26);

claims 7, 21, 33: receiving the set of predefined rules (col. 3, ll.46-49);

claims 8, 22: the set of predefined rules comprises at least one hardcoded rule, which was predefined without any user intervention by default executing all rules to check logic design (col. 4, ll.33-34, ll.39-40);

claims 9, 23: the set of predefined rules comprises at least one hardcoded rule, which has been modified in accordance with user instructions within system's ability of user's customization of the set of rules (col. 4, ll.15; col. 3. ll.57-60);

claims 10, 24: each predefined rule conforms to a single specified format (col. 3. ll.57-60);

claims 12, 26: the applying the set of predefined rules to the previously extracted information of the plurality of nodes in the netlist comprises: applying a plurality of sub-rules related by a logical operator (col. 4, ll.7-8) and as shown in the program code in the column 12

```
drc_create_sum_of_products(DRCS_TREE_NODE **node)
                    }
        expand all exclusive-or (XOR) operators in the tree
    convert all NAND and NOR operators using DeMorgan's inversion
            on the terms and subtrees feeding a node
    convert all stacked gates, e.g. AND(AND( . . . )) to single gates
            new_node = drc_compass_tree(node);
                    return (new_node);
```

claims 13. 27: the logical operator is selected from the group consisting of AND, NAND, NOR, XOR and OR as shown above in the example of program code on the column 12, (col. 19, ll.27-36);

claims 14, 28: the plurality of predefined rules is selected from the group consisting of electrical rules, connectivity rules, clock rules, timing closure rules, reset

rules and signal race rule as shown in the example of the program code (having variety of design rules including constraints) starting on the column 20 and continued on the column 21;

claims 15, 30, 34: outputting a violation report, the report containing the identified violations (col. 8, ll.14-17) and as shown in the example of the program code on the columns 11-12:

```
if both trees are valid
{
compare the error buffers and report each distinct
error, as well as all duplicated errors.
}
else report all errors to the user
```

claim 29: a storage medium that stores the extracted information such that it is available to a set of predefined rules within a computer hardware suitable for implementing the verifying the compliance of an initial logic design against the set of design rules applicable to the type of programmable device (col. 2, ll.34-37) as shown on the Fig. 2 (col. 3, ll.6-9);

claims 36, 47: extracting properties from the netlist and making them available for execution of the first and second design rules in (b) and (c) by applying every rule within aggressive MPLS checking (col. 4, ll.38-40) within applying every rule to each node in the hierarchical structure of the node tree (col. 19, ll.57-59), wherein DRC design rules are in the data structure (col. 17, ll.36-48);

claims 37, 48: using the generic routine to execute a plurality of additional design rules at the first node within an ability of the system to select additional rules by user and add them to the list of the design rules (col. 17, ll.53-55);

claim 38: using the generic routine to execute the first deign rule at a second node (col. 24, ll.5-7);

claim 39: using the generic routine to execute all design rules at first node, then using the generic routine to execute all design rules at a second node (col. 24, ll.8-11);

claim 40: using the generic routine to execute all design rules at a plurality of other nodes, then using the generic routine to execute all design rules at the plurality of other rules (col. 4, ll.16-18);

claims 41, 49: traversing to neighboring node when any of the design rules require extracted properties from a neighboring node within a hierarchical structure of the node tree and checking this nodes in the hierarchical order by DRC as shown above in the program code for design rule checking and looking for neighboring nodes in the hierarchical order checking all nodes going from one node to another (col. 17, ll.-28);

claim 43: the plurality of design rules is selected from the group consisting of electrical rules, connectivity rules, clock rules, timing closure rules, reset rules and signal race rule as shown in the example of the program code (having variety of design rules including constraints) starting on the column 20 and continued on the column 21;

claim 44: employing the generic routine comprises: employing a logical operator for checking a plurality of sub-rules, the plurality of sub-rules being part of a

corresponding design rule within the plurality of design rules operator (col. 4, ll.7-8) and as shown in the program code in the column 12, lines 33-39 (col. 19, ll.25-37);

claim 45: the logical operator is selected from the group consisting of AND, NAND, NOR, XOR and OR as shown above in the example of program code on the column 12, (col. 19, ll.25-37).

Allowable Subject Matter

6. Claims 4-6, 15, 18-20, 25 and 42 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims. Prior art of record does not teach extracting information for each node in the netlist only once; extracting information operation determines what information is required to apply the predefined rules to the plueality of nodes; extracted information operation comprises one or more properties selected from the group consisting of fanin count, fanout count, atom type, and port source atom type; the format of the predefined rules is hierarchical arrangement comprising a rule which includes one or more sub-rules and a sub-rule can include one or more basic-rules.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Helen Rossoshek whose telephone number is 571-272-1905. The examiner can normally be reached on 7:00-4:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Matthew S. Smith can be reached on 571-272-1907. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Examiner
Helen Rossoshek
AU 2825

A. M. Thompson
Primary Examiner
Technology Center 2800